

임베디드 리눅스 개발도구 기술동향

노영욱*, 변정용**, 이정배***

● 목 차 ●

1. 서론
2. 리눅스와 임베디드 시스템
3. 임베디드 리눅스 개발동향
4. 결론

1. 서론

시장 조사 기관들은 2002년 이후에 컴퓨터 산업에서 PDA, HPC, 스마트폰, 셋톱박스, 정보가전 등과 같은 포스트 PC 시장 규모가 PC 시장을 앞설 것이라는 전망을 하고 있다. 이와 같은 전망에 따라 최근에 이들 제품의 운영체제로 사용되는 임베디드 운영체제에 대한 관심이 높아지고 있다. 임베디드 시스템은 원래 <표 1>과 같은 응용 분야에서 많이 사용되어 왔다. 이러한 임베디드 운영체제 개발의 주요 내용으로는 component화 기술, down size, windows 기능 제공, realtime Java solution, 통신 프로토콜 스택 제공 등이 있다.

최근에 임베디드 시스템의 운영체제로 리눅스를 사용하려는 임베디드 리눅스에 대한 연구가 많이 이루어지고 있다. 국내외에서 임베디드 리눅스를 이용한 기술 개발 현황으로는 컴팩과 아젠다에서는 PDA를 개발중이며, 에릭슨과 노키아에서는 웹폰과 휴대폰을 개발 중이다. 국내에서는 SK 텔레

콤이 CDMA/IMT-2000 스마트폰 시제품을 발표하였고 삼성전자와 LG전자에서 포스트PC 제품을 개발 중에 있다.

IDC는 올해 임베디드 시스템에 탑재되는 소프트웨어가 약 60억달러 규모의 시장을 형성할 것이라고 전망한다. 그리고 미국의 시장 조사기관인 VDC는 임베디드 리눅스의 임베디드 소프트웨어 시장 점유율이 2003년에는 14%, 2010년에는 50%로 성장할 것이라는 예측을 하였다. 따라서 리눅스를 임베디드 시스템의 운영체제로 사용할 시장 규모는 크고, 유망한 분야이다.

현재 국내외에서 많이 사용되는 VRTX, pSOS, VxWorks 등의 실시간 운영체제(RealTime OS)는 개발자 한 명당 라이선스 가격이 비싼 것은 3000만원 정도이다. 따라서 상용화된 실시간 운영체제를 이용한 각종 장비의 개발 비용이 높아져서 제품의 가격 경쟁력을 저하시킨다. 즉, 운영체제의 로열티 비용에 대한 부담으로 제품의 가격 경쟁력을 저하시키고 있다. 그러나 임베디드 리눅스를 기반으로 한 시스템 소프트웨어는 저렴한 가격, 공개된 소스 코드, 안정성, 기능성에서 우월하여 품질 경쟁력을 향상하고 로열티 부담을 크게 줄여서 가격 경쟁력을 높일 수 있다. 또한 리눅스는 다양한 시스템 구

* 신라대학교 컴퓨터교육과 조교수

** 동국대학교 교수

*** 부산외국어대학교 부교수

<표 1> 주요 임베디드 응용 분야

장치 분류	사업용 응용 예	계산 모델	회사
Smart Mobile과 Handheld 통신장비	• PDA, cellular phones	• 제한된 디스플레이와 메모리 크기	• Nokia
자동차시스템	• GPS • 오락용 • 웹서버	• Fat client • Fat server	• Daimler-Benz • BMW • Microsoft/Netscape
가정용 응용	• Net TVs와 set-top boxes • Screen phones • 가전 NCs	• Fat client • Thin client	• Microsoft/WebTV • infoGear
사무 시스템	• 복사기, 프린터, 팩스	• Thin server	• Hewlett-Packard • Tektronix
통신과 네트워크 시스템	• 교환기 • 라우터와 게이트웨이	• Thin server	• Cisco
분산 센서 시험과 측정 시스템	• 시험과 측정 장비 • 네트워크 카메라 • 대규모 센서	• Thin 서버/ 클라이언트 • Peer-to-peer IP Multicast	• Hewlett-Packard Axis Communications

성 방안과 구현 방법이 있으므로 다양한 접근 방법으로 연구 개발이 가능하다. 본 고에서는 최근의 각종 임베디드 리눅스 소프트웨어 관련된 동향을 살펴본다.

2. 리눅스와 임베디드 시스템

임베디드 시스템 소프트웨어의 개발 환경은

cross 개발환경과 native 개발환경으로 분류 할 수 있다. Cross 개발환경은 UNIX나 Windows에 개발환경을 구축하고 이곳에서 특정 목표 보드(target board)에 맞는 실시간 운영체제 및 응용 프로그램 이미지를 이 기종간에 컴파일(cross 컴파일)하여 target 보드에 전송하여 수행하는 방식이다. Native 개발환경은 호스트와 목표의 운영체제가 같은 경우를 말한다. <표 2>는 대표적인 임베디드 운영체

<표 2> 임베디드 운영체제 종류

제품명	회사	특징	최소크기	개발호스트	개발환경
VxWorks	Windriver	• 확장 가능한 마이크로커널 구조 • 다중프로세서 지원 • POSIX 호환성	5 KB	PC, Sun, HP, SGI, IBM	Cross
pSoS	Intergrated Systems (WindRiver에 인수)	• 확장가능한 마이크로커널구조 • 다중프로세서 지원 • 다중 객체 대기 지원	2.5 KB	PC, Sun	Cross
OS-9	Microware	• 68 ~ 308KB • 제한된 기능에서 고성능을 제공하는 작은 실시간 운영체제	68 KB	PC, Unix, native	Native
QNX	QNX Systems	• 작고 안정적인 GUI 지원 • 작고 빠른 마이크로커널 구조	.	PC, Unix	Native
VRTX	Microtech	• 다양한 CPU 지원 • 결정적, 선점적 멀티태스킹 • 모듈화, 라이브러리 기반 구조	16 KB	PC, Sun	Cross
Windows CE	Microsoft	• 최대 2MB 크기 • 가장 크고 가장 느린 모빌 OS • 다양한 장치 지원 • 윈도우 기능 제공	.	.	Cross
Epoc	Symbian	• 1MB • 개방형 platform	.	.	Native

제의 종류와 특징을 요약한 것이다.

상용화된 VxWorks, pSOS, VRTX, OS-9, QNX, WinCE, Embedded Java 등을 임베디드 시스템의 운영체제로 사용할 경우에 다음과 같은 문제점이 있다.

- 크기가 너무 크고 커널의 재구성이 쉽지 않다.
- 임베디드 휴대용 장비가 지원하는 자원의 제약을 극복하기가 어렵다.
- 다양한 시스템을 구성하기가 힘들다.
- 초기 구입비와 로열티가 너무 비싸다.
- 개발 인력이 많지 않다.

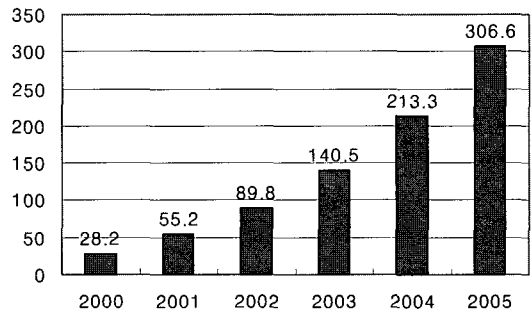
2001년에 Evans Data Corporation에서는 임베디드 시스템 분야에서 개발자로 적극적으로 활동하는 500여명의 개발자(이 중에 85%는 북미에 거주)를 대상으로 사용하기 희망하는 임베디드 운영체제를 조사하였다. 이 결과에 의하면 2001년도에는 임베디드 리눅스가 VxWorks, MS-DOS에 이어서 3번째로 많이 사용하기를 희망하나 2002년에는 가장 많이 사용될 것으로 예상하고 있다. <표 3>은 Evan Data Coporation에서 조사한 상위 10위의 임베디드 운영체제를 순위별로 나타낸 것이다.

<표 3> 임베디드 시스템 개발자의 희망 목록[1]

순위	2001년	2002년
1	Home grown OS와 커널	Embedded Linux
2	WinderRive OS (VxWorks and VxWorks AE)	WinderRive OS
3	DOS	Windows CE
4	Embedded Linux	LynxOS
5	Windows CE	BSD
6	QNX	Green Hills
7	ATI Nucleus	QNX
8	Green Hill OS(ThreadX and Integrity)	DOS
9	LynuxWorks LynxOS	Chorus
10	Sun MicroSystems Chorus	Home grown

그리고 임베디드 시스템 시장에서 리눅스의 미래에 대한 VDC의 분석 자료[2]에 의하면, (그림 1)과 같이 2000년에 전세계적으로 임베디드 리눅스

운영체제, 소프트웨어 개발 도구와 관련된 서비스의 시장 규모가 약 \$2,800,000이나 2005년에는 \$306,600,000에 달해 연 평균적으로 61.2%로 성장할 것으로 예측하고 있다. 이와 같이 임베디드 시스템 분야에 임베디드 리눅스가 많이 사용되는 이유는 리눅스의 안전성과 성장 가능성에서 원인을 찾을 수 있다.



(그림 1) 전세계 임베디드 리눅스 시장 규모[2]

상용화된 실시간 운영체제의 문제점을 해결하기 위해 임베디드 리눅스를 인터넷 정보가전의 운영체제로 사용할 경우에 해결해야 할 기술적인 과제는 다음과 같다[3,4].

• 저전력 시스템 개발

소형 휴대 정보 단말기의 개발에 있어 효율적인 전력 관리를 통한 배터리 사용 시간의 연장은 중요한 경쟁력 향상의 요소이다. 리눅스 커널에 이용되고 있는 진보된 전력 관리 기술을 연구 개발하고 이를 소형 시스템에 구현하는 일이 필요하다.

• 효율적인 메모리 관리 기술

제한된 메모리를 갖는 소형 시스템에 다양한 기능을 갖는 소프트웨어를 지원하여 성능을 향상시키는 기술이 필요하며, 리눅스에 이용된 메모리 및 프로세스 관리 기술을 연구 개발하여 이를 소형 시스템에 구현하여야 한다. 특히 실행 후 잔류 메모리를 감소시키는 연구가 필요하다.

• 사용자 인터페이스 연구 및 개발

사용자 인터페이스를 필요로 하지 않는 임베디

드 시스템도 있지만, PDA를 포함한 많은 임베디드 시스템들은 간단한 텍스트 기반에서부터 훨씬 복잡한 방식까지 사용자 인터페이스를 필요로 한다. PDA는 키패드, 필기체 인식, 음성인식 등이 연구 개발되어 왔으며, 최근에는 한 손만으로 사용할 수 있는 방향성이나 무게성에 기반을 둔 사용자 인터페이스에 대한 개발도 이루어지고 있다. 사용자 편의성과 전달의 정확성을 중시하는 새로운 개념의 사용자 인터페이스에 대한 개발은 리눅스 뿐만 아니라 대부분의 임베디드 시스템의 연구 분야이다.

- 안정적 시스템 소프트웨어 기술 개발

데스크탑 컴퓨터에 쓰이는 리눅스는 다른 운영체제에 비해 향상된 안정성을 갖고 있다. 특히 통신 구성요소에서의 리눅스의 안정성을 소형 시스템에도 그대로 유지할 수 있는 기술 개발이 필요하다.

- 실시간성

휴대 정보 단말기를 멀티미디어 플레이어로도 사용하기 위해서 가장 필요한 요소 중의 하나는 운영체제의 실시간성이다. 연성 실시간성은 기존의 리눅스에 이미 POSIX 연성 실시간 스케줄링으로 구현되어 있으며, hard 실시간성을 커널에 구현하는 연구는 최근에 이루어지고 있다. 실시간 리눅스는 멀티미디어 단말 뿐만 아니라 네트워크 라우터 장비에도 적용되고 있어 실시간성에 대한 연구가 필요하다.

- 부분 시스템으로서의 임베디드 시스템

PDA와 같은 임베디드 시스템 및 이동 단말은 작고 가벼워야 하므로 컴퓨팅 파워 및 메모리 사이즈에 있어서 제한적이고 하드 디스크 같은 대용량 저장 장치를 장착하기도 어렵다. 그리고 휴대 장비는 단말 자체가 독립적인 완전한 시스템으로 작동하여야 하기 때문에 이 부분에 대한 연구 개발이 필요하다.

3. 임베디드 리눅스 개발 동향

임베디드 리눅스 장치를 위한 소프트웨어를 개발할 때 사용할 수 있는 다양한 특징을 갖는 제품과 프로젝트는 여러 가지가 있다. [5]에서는 “임베디드 리눅스 툴킷”을 호스트 리눅스 시스템과 목표 하드웨어를 가지고 리눅스를 부팅하고 간단한 응용을 수행할 수 있는 하드웨어에 적재할 이미지를 만들 수 있는 완전한 해결책을 제공하는 것으로 정의한다. 툴킷마다 다른 사용자 인터페이스와 스타일을 가지나 본질적으로 같은 일을 해야한다. 즉, 단일 커널 쓰레드로 응용을 포함하는 리눅스 커널을 생성하거나 루트 파일시스템 내에 응용을 포함하도록 하는 루트 파일시스템을 생성한다. 툴킷은 목표 시스템이 부트하는 과정에서 binary 이미지를 조각 형태로 설치할 수 있어야 한다. 그러나 여기서는 [6]에서 분류한 기준에 따라 현재 연구 개발된 임베디드 리눅스 소프트웨어를 <표 4>와 같이 정리하였다. 이 이외에도 많은 소프트웨어들이 있으며 각 소프트웨어 대한 구체적인 사양과 특징은 [6]과 <표 4>의 사이트를 참조하기 바란다. 이 중에 BlueCat Linux, PeeWeeLinux, Midori와 BusyBox에 대해 살펴본다.

3.1 BlueCat Linux

BlueCat Linux는 크로스 개발 환경에서 임베디드 시스템에 사용하기 위한 LynuxWorks의 리눅스 배포판이다. BlueCat은 전자통신, 네트워크 제품, 군사 장비, 의료 장비와 고성능 프린터 및 복사기와 같은 목표 시스템에서 사용될 임베디드 응용, 커널과 장치 구동기의 생성을 지원한다. BlueCat은 임베디드 시스템에서 사용될 소스 이미지를 만들기 위한 개방형 소스 커널, GNU/EGCS의 개발 도구, C와 C++ 컴파일러, 디버거, 프로파일러, 라이브러리와 도구들을 포함하고 있다. 커널 디버거는 목표 시스

<표 4> 임베디드 리눅스 소프트웨어[6]

종류	이름	site	
임베디드 리눅스 배포판	BlueCat	http://www.lynx.com/	
	Embedix	http://www.lineo.com/	
	Emj-Linux	http://www.emjembedded.com/	
	ET-Linux	http://www.prosa.it/etlinux/	
	HardHat Linux	http://www.mvista.com/products/hardhat.html	
실시간 리눅스	QLinux	http://lass.cs.umass.edu/software/qlinux/	
	Realtimelinux.org	http://www.realtimelinux.org/	
	RTAI	http://www.aero.polimi.it/projects/rtai/	
	RT-Linux vs standard Linux	http://www.fisica.unlp.edu.ar/rt/	
	Zentropix	http://www.lineo.com/products/embedix_realtime/index.html	
임베디드 리눅스 도구	윈도우 환경	Dinx	http://dinx.sourceforge.net/
		FLTK	http://www.fltk.org/
		Microwindows	http://microwindows.censoft.com/
		Minigui	
		MGR	http://www.ibiblio.org/pub/Linux/apps/MGR/!INDEX.html
	유틸리티	ae	http://www.debian.org/Packages/unstable/base/ae.html
		ash	http://packages.debian.org/unstable/shells/ash.html
		Busybox	http://www.busybox.net/
		Bzexe	http://linux-embedded.com/bzexe
		elvis-tiny	http://packages.debian.org/unstable/base/elvis-tiny.html
		Pump	
		TinyLogin	
	UPX	http://wilsau.idv.uni-linz.ac.at/mfx/moved-upx.html	
	Network	Envoy	http://www.icon-labs.com/
		Etherboot	http://etherboot.sourceforge.net/
		iproute	http://packages.debian.org/unstable/net/iproute.html
		Netboot	http://netboot.sourceforge.net/
	기타	Nilo	http://www.welcomehome.org/nilo/
		Opera	http://www.opera.com/
		Syslinux	ftp://ftp.kernel.org/pub/linux/utils/boot/syslinux/
	Yard	http://www.linuxlots.com/~fawcett/yard/	
작은 리눅스 배포판	Alfalinix, Ballantain, BYLD, cclinux, Coyote, DosLinux, Dialup Routeur Mini, DLX, DragonLinux, Emblin, Empire, Fluf, Floppyfw, Green Frog, hal91, Iron Wing, Latrell Sprewell, LEM, Linux Router Project, Live CD, Loaf, Loop, LxA, Metalab, mulinux, PeeWeeLinux, Pocket Linux, Pygmy, Small Linux, Subliminal, ThinLinux, Tiny, Tomsrtbt, Traveller's, Trinix, Xdenu		

템에서 사용자 드라이버 개발을 지원하며, 사용자는 LynuxWorks 유틸리티를 사용하여 부팅 가능한 디스크와 ROM을 생성할 수 있다. BlueCat은 여러 장의 CD로 배포되며 모든 소스 코드를 담고 있는 한 장의 CD와 각 아키텍처의 바이너리별로 한 장의 CD가 있고 Windows와 리눅스 호스트에서 크로스 개발을 지원한다. BlueCat은 C 라이브러리로 glibc를 사용하므로 2MB 이하의 저장 공간을 갖는 임베디드 장치에 많은 기능을 넣기가 어려운 점이 있다. BlueCat Linux 4.0의 특징은 다음과 같다[5,7].

- 2.4 Linux 커널을 기반으로 GNU 2.95.3 도구들 지원한다.
- 다양한 하드웨어 구조 지원한다.
 - IntelR XScale™ microarchitecture, IXP1200
 - Network Processors and IA-32
 - MIPS
 - ARM family (including Thumb extensions)
 - PowerPC
 - x86 PC-compatibles
- 리눅스와 윈도우 호스트 플랫폼을 위한 향상

된 리눅스 크로스 개발 환경 지원한다.

- 다양한 개발 도구를 지원한다.
 - BusyBox, TinyLogin
 - JTAG와 Watchdog 타이머 지원
 - gdb와 gcc 및 향상된 GNU 크로스 개발 도구
 - 멀티 쓰레드화된 디버거 및 커널 디버깅 기능
- Zebra 라우팅 프로토콜 패키지와 향상된 네트워크 관리와 보안 특징 제공
- 저널링 파일시스템 제공

3.2 PeeWeeLinux

PeeWeeLinux는 1999년 Adi Linden에 의해 만들어졌으며 현재 Adi Linden에 의해 유지되고 있다. PeeWeeLinux는 임베디드 응용과 단일 플로피 시스템을 위한 작은 리눅스 시스템을 쉽게 생성하기 위해 개발되었다. PeeWeeLinux의 기본적인 설계 목표는 종속성을 검사하여 특정 응용에 요구되는 모든 라이브러리와 지원 파일을 PeeWeeLinux binary 패키지에 포함시키고 나머지는 제거하여 binary 패키지를 최소화한다. 모든 binaries와 라이브러리는 분리되고 PeeWeeLinux에서는 PAM를 제공하지 않으며 터미널은 termcap 라이브러리를 사용하여 지원되며 현재 ncurses 라이브러리는 지원되지 않는다.

PeeWeeLinux 배포판은 RedHat 리눅스 플랫폼에서 개발되었으며, 현재는 소스로부터 PeeWeeLinux를 컴파일하기 위해서는 RedHat 리눅스가 필요하나 생성된 binary 배포판은 독립적으로 배포할 수 있다. PeeWeeLinux는 임베디드 플랫폼에서 가능한 쉽고 고통없이 리눅스 운영체제를 구성하고 설치하도록 하는 환경을 제공하기 위해 개발 중에 있다. 구성 유틸리티는 메뉴 방식을 사용하여 완전한 패키지나 패키지로부터 파일의 이 부분을 목표시스템에 포함되도록 할 수 있다. syslinux나 lilo 부트로더를 사용하는 목표 시스템이 지원되며, 목표 시스템은 루트 ramdisk, read-only 루트 파티션과 일반적

인 단일 read-write 루트 파티션으로 구성할 수 있다. PeeWeeLinux로 생성 가능한 응용으로는 복구용 플로피 디스크, 라우터, 방화벽, Thin-클라이언트 등이 있다[5,8].

PeeWeeLinux의 주요 특징은 다음과 같다.

- RedHat 6.2 플랫폼에서 개발
- rpm을 사용하여 패키지를 생성하고 유지
- 메모리 footprint를 최소화하기 위해 패키지를 개별화 가능
- Ncurses을 이용한 그래픽 구성과 설치 도구 제공
- 임베디드 응용을 위해 향상된 2.2.x 커널 사용
- USB, PCMCIA 지원
- XFree86 지원

3.3 미도리(Midori)

미도리는 인터넷 정보 가전(interbet appliance)과 같은 리눅스 기반의 장비에서 응용 디바이스의 전력의 효율을 향상시키기 위한 시스템 소프트웨어의 모음이다. 미도리와 그 소프트웨어 개발 환경은 오픈 소스 소프트웨어와 하드웨어 요소 장비의 개발을 기반으로 하고 있으며, 임베디드 소프트웨어 응용을 위해 설계된 만들어진 패키지 관리 시스템을 가지고 있다. 미도리는 Windows CE나 PalmOS 처럼 monolithic한 플랫폼이 아니며 Debian이나 Mandrake처럼 상업용 데스크탑 리눅스 배포판으로 완벽하지도 않으나 임베디드 제품을 개발하는데 유용하게 쓰여질 기술의 집합이다[9].

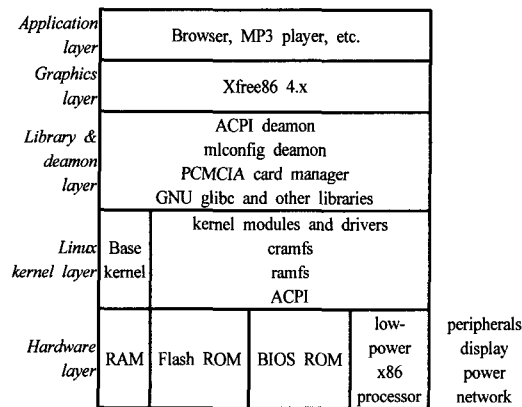
미도리는 정보 가전의 특징인 저전력, 빠른 부팅, 제한된 용량의 비휘발성 저장장치, 인터넷 연결성, 이동성, 쉬운 사용법, 개인용 장비지원, 네트워크 기반의 클라이언트 관리를 지원하기 설계된 소프트웨어 기술이다.

일반 리눅스 커널과 호환하기 위해 ACPI 인터페이스를 사용하여 전력 관리를 위해 노력하였으며, Cramfs(compressed memory file system)은 compact

flash memory 카드로부터의 정보의 저장능력 및 인출량을 늘려주었다. Cramfs는 같은 형식의 다른 시스템과 API를 다른 리눅스와 마찬가지로 지원하며 라이브러리와 프로그램들 및 리소스(폰트, 그래픽, 아이콘, 미디어 파일 등)를 추가함으로써 과정을 단순화하였다. 이는 단순히 파일을 지금 사용되어지는 디렉토리 내에서 복사함으로써 이루어진다. Build system은 packcramfs를 사용하여 파일들을 대상 디바이스의 Flash ROM 이미지로 옮긴다. Ramfs는 수행시간 기반의 정보를 저장하기 위한 RAM-based 파일 시스템이다.

미도리는 (그림 2)와 같이 Linux 2.4 커널의 일반 데스크탑 환경 리눅스와 매우 유사한 구조를 가지나 일반 리눅스와는 다음과 같은 차이가 있다.

- Flash ROM으로부터의 부팅
- 디스크가 필요 없는 오퍼레이션
- Script-based init
- No login prompt
- 제한된 유틸리티



(그림 2) Midori 구조

3.4 BusyBox

BusyBox는 Erik Andersen에 의해 관리되며 GNU GPL에 따라 라이선스 되는 것으로 매우 작은 공간에 많은 소프트웨어를 집어넣을 수 있는 것으로 특히 정적으로 링크된 환경에서 유용하다. BusyBox

는 UNIX에서 많이 사용되는 유틸리티들을 간소화하여 하나의 작은 수행 파일에 결합한 것이다. GNU의 fileutils, shellutils, findutils, textutils, grep, gzip, tar 등에 있는 유틸리티를 기능을 간소화하여 몇 개의 옵션 만이 제공한다. 그러나 유틸리티에서 제공하는 옵션은 대부분의 환경에서 기대하는 기능을 완전히 제공한다. BusyBox는 원래 Debian 복구와 설치 디스크를 지원하기 위해 작성되었으나 현재는 작거나 임베디드 환경을 위한 완전한 POSIX 호환을 제공한다[5,10].

Busybox는 일반적으로 gcc에 의해 지원되는 구조에서 생성되므로 `_sparc_` 과 `_alpha_` 를 위한 몇 개의 추가적인 특징이 있으며, `insmod` 기능은 현재 x86, ARM, SH3/4, powerpc, m68k과 MIPS에서만 가능하다. 지원되는 libcs로는 `glibc-2.0.x`, `glibc-2.1.x`, `glibc-2.2.x`, `Linux-libc5`, `uClibc`이 있으며, 지원되는 커널로는 `Linux 2.0.x`, `2.2.x`, or `2.4.x`이 있다. 현재 BusyBox에서 정의된 함수는 <표 5>와 같다.

<표 5> BusyBox에서 정의된 함수

BusyBox에서 정의된 함수
adjtimex, ar, basename, busybox, cat, chgrp, chmod, chown, chroot, chvt, clear, cmp, cp, cpio, cut, date, dc, dd, deallocvt, df, dirname, dmesg, dos2unix, dpkg, dpkg-deb, du, dumpkmap, dutmp, echo, expr, false, fbset, fdflush, find, free, freeramdisk, fsck.minix, getopt, grep, gunzip, gzip, halt, head, hostid, hostname, id, ifconfig, init, insmod, kill, killall, klogd, length, ln, loadacm, loadfont, loadkmap, logger, logname, ls, lsmod, makedevs, md5sum, mkdir, mkfifo, mkfs.minix, mknod, mkswap, mktemp, more, mount, mv, nc, nslookup, ping, pivot_root, poweroff, printf, ps, pwd, rdate, readlink, reboot, renice, reset, rm, rmdir, rmmmod, route, rpm2cpio, sed, setkeycodes, sh, sleep, sort, stty, swapoff, swapon, sync, syslogd, tail, tar, tee, telnet, test, tftp, touch, tr, true, tty, umount, uname, uniq, unix2dos, update, uptime, usleep, uudecode, uuencode, watchdog, wc, wget, which, whoami, xargs, yes, zcat, [

4. 결론

임베디드 리눅스는 임베디드 시스템의 운영체제로 가장 많이 사용될 것으로 예상되고 있다. 임베디드 리눅스에 대한 연구는 <표 4> 이 외에도 많은 연구 개발이 진행 중에 있으나 해외의 활동과 비교할 때 국내에서의 활동은 부족한 점이 많다. 향후 임베디드 리눅스가 많이 사용되기 위해서는 리눅스 커널, 장치 구동기, 응용 프로그램 및 GUI 개발자의 확보와 윈도우에 기반한 개발 환경의 일반화가 필요하다. 이를 위해 각 학교와 교육 기관에서 임베디드 리눅스에 대한 교육을 확대하고 기업에서는 임베디드 리눅스를 이용한 수익 모델을 찾는 것이 아주 중요한 과제라고 생각한다.

참고문헌

- [1] Embedded Linux tops developers' 2002 wishlist, LinuxDevices.com, 2001. 17
- [2] Stephen Balaco, Whitepaper: Linux's Future in the Embedded Market, LinuxDevices.com, 2001.6.
- [3] Rick Lehrbaum, "Using Linux in Embedded and Real-time Systems," LinuxDevices.com, 2000.2.
- [4] 박영준, 리눅스 플랫폼의 임베디드 프로그래밍, 프로그램 세계, 2000.7.
- [5] John Lombardo, Embedded Linux, New Riders, 2002.
- [6] <http://linux-embedded.com/software.php3>
- [7] BlueCat Linux Users Guide - Release 4.0, LynuxWorks, 2001.
- [8] <http://www.peeweelinux.org/Documentation/PeeWeeLinux.html>.
- [9] <http://midori.transmeta.com/>
- [10] <http://www.busybox.net/>

저자약력



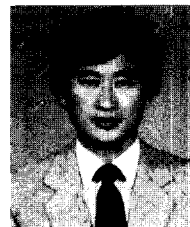
노영욱

1985년 부산대학교 계산통계학과(학사)
 1989년 부산대학교 대학원 전자계산학과(석사)
 1998년 부산대학교 대학원 전자계산학과(박사)
 1989년-1996년 한국전자통신연구원 연구원
 1996년-현재 신라대학교 컴퓨터교육과 조교수
 관심분야: 운영체제, 멀티미디어, 병렬/분산시스템, 컴퓨터교육
 e-mail : yulho@silla.ac.kr



변정웅

1980년 동국대학교 컴퓨터학과(학사)
 1983년 동국대학교 컴퓨터학과(석사)
 1994년 홍익대학교 컴퓨터학과(박사)
 1982년-1988년 한국전자통신연구원 연구원
 1995년-1996년 영국 요크대학교 방문교수
 1983년-현재 동국대학교 교수
 관심분야: 실시간 DB, 정보 검색, 한글 부호화



이정배

1981년 경북대학교 전산공학과(공학사)
 1983년 경북대학교 전산공학과(공학석사)
 1995년 한양대학교 전산공학과(공학박사)
 1982년-1991년 한국전자통신연구원 선임연구원
 1996년-1997년 U.C.Irvine 객원교수 Dept.of Electrical & Computer Eng.
 1991년-현재 부산외국어대학교 컴퓨터공학과 부교수
 관심분야: 컴퓨터네트워크, 실시간시스템, 인터넷 응용